

PRINT SYSTEM

Publication number: JP2002014791

Publication date: 2002-01-18

Inventor: MORITA TAKEHIKO

Applicant: RICOH KK

Classification:

- international: **B41J29/38; G06F3/12; B41J29/38; G06F3/12; (IPC1-7):**
G06F3/12; B41J29/38

- european:

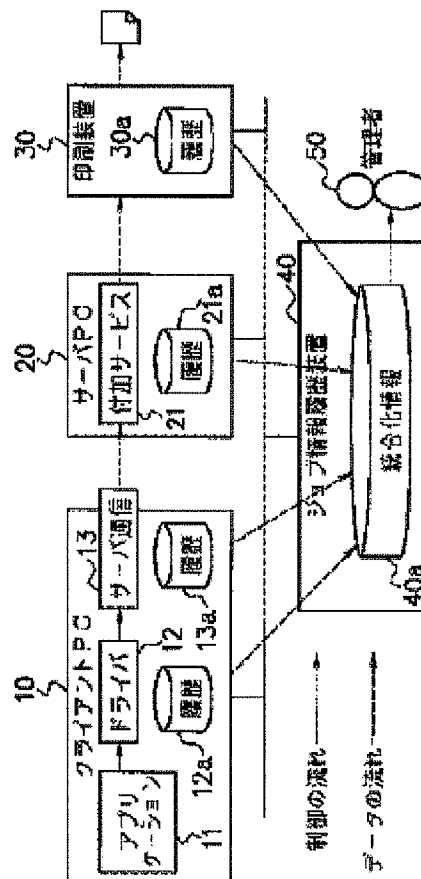
Application number: JP20000353484 20001120

Priority number(s): JP20000353484 20001120; JP20000123237 20000424

Report a data error here

Abstract of JP2002014791

PROBLEM TO BE SOLVED: To provide a print system which integrates a series of pieces of information regarding a print job and provides the whole image of the integrated job for a user, with respect to a print system which is connected to a network. **SOLUTION:** A job history information device 40 collects job history information stored in respective modules. Arbitrary history information in the collected history information is regarded as reference history information and one of jobs constituting the reference history information is decided as a reference job. Job information on a job included in history information different from the reference history information judged to be done by the same print job as the reference job is added to job information on the reference job. After the job information is added, similar operation is carried out as to the job of other history information and this operation is carried on for jobs of all history information. After the operation is completed for the jobs of all the history information, different history information is regarded as reference history information and the same operation is carried out.



Data supplied from the esp@cenet database - Worldwide

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2002-14791

(P2002-14791A)

(43) 公開日 平成14年1月18日 (2002.1.18)

(51) Int.Cl. ⁷	識別記号	F I	テーマコード* (参考)
G 0 6 F 3/12		G 0 6 F 3/12	D 2 C 0 6 1
			A 5 B 0 2 1
B 4 1 J 29/38		B 4 1 J 29/38	Z

審査請求 未請求 請求項の数14 O L (全 14 頁)

(21) 出願番号 特願2000-353484 (P2000-353484)
(22) 出願日 平成12年11月20日 (2000. 11. 20)
(31) 優先権主張番号 特願2000-123237 (P2000-123237)
(32) 優先日 平成12年4月24日 (2000. 4. 24)
(33) 優先権主張国 日本 (J P)

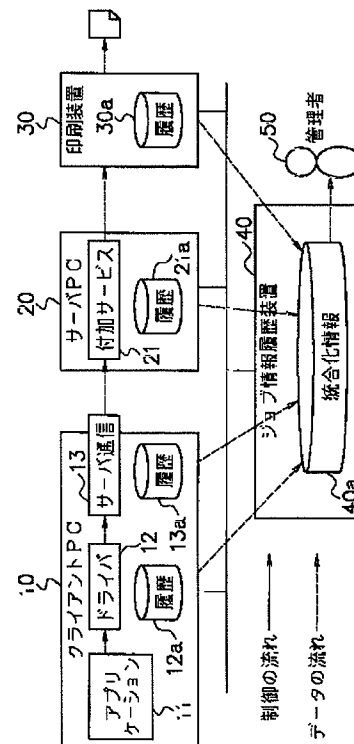
(71) 出願人 000006747
株式会社リコー
東京都大田区中馬込1丁目3番6号
(72) 発明者 森田 岳彦
東京都大田区中馬込1丁目3番6号 株式
会社リコー内
Fターム (参考) 2C061 AP01 HJ10 HQ12 HQ17
5B021 AA01 AA02 BB10 CC04 EE04

(54) 【発明の名称】 印刷システム

(57) 【要約】

【課題】 ネットワークに接続される印刷システムにおいて、印刷ジョブに関する一連の情報を統合化し、統合化されたジョブの全体像をユーザに提供する印刷システムを提供する。

【解決手段】 ジョブ履歴情報装置40は、各モジュールに保存されるジョブ履歴情報を収集する。収集された履歴情報の内の任意の履歴情報を基準履歴情報にして、基準履歴情報を構成するジョブの内の1つを基準ジョブとする。基準ジョブと同じ印刷ジョブで行われたと判断される基準履歴情報とは別の履歴情報に含まれるジョブのジョブ情報を基準ジョブのジョブ情報に追加する。追加したらさらに別の履歴情報のジョブについても同様の動作を行い、全ての履歴情報のジョブについて終了するまでこの動作を続ける。全ての履歴情報のジョブについて行ったら、さらに別の履歴情報を基準履歴情報にして同じ動作を行う。



【特許請求の範囲】

【請求項1】 ネットワーク上のクライアントの有する印刷データ生成部が印刷データを前記ネットワークを通じて送信して印刷する印刷システムであって、印刷装置駆動部は、ドライバ履歴情報を記憶する第1の記憶手段を有し、印刷装置は、前記印刷装置が行った印刷装置履歴情報を記憶する第2の記憶手段を有し、印刷管理装置は、前記第1の記憶手段が記憶するドライバ履歴情報と、前記第2の記憶手段が記憶する印刷装置履歴情報とを読み出す手段と、所有するデータを前記クライアントに転送する転送手段と、を有することを特徴とする印刷システム。

【請求項2】 前記印刷管理装置は、さらに、前記読み出し手段によって読み出された前記ドライバ履歴情報および前記印刷装置履歴情報を記憶する第3の記憶手段を有することを特徴とする請求項1に記載の印刷システム。

【請求項3】 前記印刷管理装置は、読み出された前記ドライバ履歴情報および前記印刷装置履歴情報のいずれかに含まれるデータに示される処理を選択し処理を決定する基準ジョブ決定手段と、該基準ジョブ決定手段によって決定された基準の処理とは別のデータで示される処理が同一の前記印刷ジョブで行われているか否かを判断する判断手段と、前記基準の処理を示すデータとは別の処理を示すデータを基準の処理を示すデータに追加するデータ追加手段と、を有し、前記基準の処理とは別のデータで示される処理が前記判断手段によって前記印刷ジョブで行われていると判断された場合、前記データ追加手段によって追加されることを特徴とする請求項1または2に記載の印刷システム。

【請求項4】 前記追加手段によって追加された別の処理を示すデータを、前記追加手段によって追加される前に格納されていた前記ドライバ履歴情報および前記印刷装置履歴情報のいずれか1つを削除する第1の削除手段を有することを特徴とする請求項3に記載の印刷システム。

【請求項5】 前記印刷データ生成部は、前記ネットワーク上および該ネットワークに接続される端末内で共通で、各々の前記印刷ジョブによって区別される識別情報を前記印刷データに組み込む識別情報組み込み手段を有し、前記判断手段による判断の基準は、前記識別情報組み込み手段によって組み込まれた識別情報に基づくことを特徴とする請求項3または4に記載の印刷システム。

【請求項6】 前記印刷管理装置は、前記ドライバ履歴情報および前記印刷装置履歴情報のいずれか1つに含まれる1つの情報を構成する1つの要素である情報項目を削除する第2の削除手段を有し、

前記判断手段によって判断される際に、前記ドライバ履歴情報の情報項目と前記印刷装置履歴情報の情報項目とが一致する場合、前記ドライバ履歴情報の情報項目および前記印刷装置履歴情報の情報項目のいずれか一方を前記第2の削除手段によって削除することを特徴とする請求項1から5のいずれか1項に記載の印刷システム。

【請求項7】 前記ネットワークにはさらにサーバが接続され、前記サーバは前記印刷ジョブにおいて前記サーバがどのような処理を行ったかというサーバ履歴情報を記憶し、保存する第4の記憶手段を有し、前記読み出し手段は、前記第4の記憶手段が記憶するサーバ履歴情報をさらに読み出すことを特徴とする請求項1に記載の印刷システム。

【請求項8】 前記ネットワークにはさらにサーバが接続され、前記第3の記憶手段は前記サーバ履歴情報をさらに記憶することを特徴とする請求項2に記載の印刷システム。

【請求項9】 前記ネットワークにはさらにサーバが接続され、前記基準ジョブ決定手段の選択の対象は前記サーバ履歴情報に含まれる任意の1つのデータに示される処理も含まれることを特徴とする請求項3または5に記載の印刷システム。

【請求項10】 前記ネットワークにはさらにサーバが接続され、前記第1の削除手段の削除の対象には前記サーバ履歴情報も含まれることを特徴とする請求項4に記載の印刷システム。

【請求項11】 前記ネットワークにはさらにサーバが接続され、前記第2の削除手段による削除の対象には前記サーバ履歴情報に含まれる1つのデータを構成する1つの要素である情報項目も含まれることを特徴とする請求項6に記載の印刷システム。

【請求項12】 複数のサブシステムを有する印刷システムであって、

該印刷システムは、各印刷ジョブを識別するジョブ識別子と、前記印刷システムの所定の範囲内で前記ジョブ識別子を検索するために用いられるキーとを対応づけて保存するジョブID一覧保持部を有し、前記複数のサブシステムは、前記キーによって前記識別子を検索し、該識別子によって、前記印刷ジョブにおいて前記複数のサブシステム各々で行われるジョブである断片ジョブをまとめることを特徴とする印刷システム。

【請求項13】 前記ジョブID一覧保持部は、前記ジョブ識別子を破棄する識別子破棄手段と、該破棄する時期を決める破棄時期決定手段と、をさらに有し、

前記破棄する時期は前記複数のサブシステムの通知によって決定されることを特徴とする請求項12記載の印刷システム。

【請求項14】 前記複数のサブシステムは、該サブシステムにおける前記断片ジョブの開始と終了とを前記ジ

ジョブID一覧保持部に通知するジョブ開始終了通知手段をさらに有し、

前記サブシステムは、前記ジョブ開始終了通知手段が前記断片ジョブの終了を通知してから、前記断片ジョブをまとめることを特徴とする請求項12または13記載の印刷システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、クライアントから送信された印刷データが、ネットワークを経由して、印刷サーバあるいは印刷装置に受信、処理され、印刷される印刷システムに関し、特に、各モジュールの印刷ジョブの履歴を取得、管理する印刷システムに関する。

【0002】

【従来の技術】ネットワークを経由して複数のクライアント装置からデータを送信し、ネットワークに接続される印刷装置あるいは印刷サーバにより送信されたデータを受信して処理し、印刷を行う印刷システムが実用化されている。

【0003】上記印刷システムにおいて、「印刷ジョブの履歴」を取得する方法として、例えばクライアント上の印刷ドライバが生成するジョブの情報を印刷ドライバ自体がデータとして保管する技術、印刷サーバがサービスする印刷ジョブの情報を印刷サーバ自体が保管する技術、および印刷装置が印刷するジョブの情報を印刷装置自体が保管するという技術が知られている。

【0004】このような従来技術として、特開平10-283131号公報には、印刷履歴情報をプリンタ内の記憶媒体に蓄積し、蓄積された印刷履歴情報を印刷したり、ホストコンピュータに送信したりすることによってユーザに表示する技術が提案されている。

【0005】

【発明が解決しようとする課題】本来、印刷ジョブは、ジョブの生成箇所「クライアント」から始まり、印刷サーバ等を経由し、最終的に印刷装置に到達して印刷データが印刷され、処理が終了するという、一連のパスを通る。

【0006】しかしながら、上記従来技術において、各印刷ドライバ、印刷サーバおよび印刷装置は、通過するパスの中のただ1点において、その点（場所）で取得し得る印刷ジョブ情報を保管するのみに限られていた。

【0007】例えば印刷ドライバは、ジョブの開始点であるクライアント上に存在するため、ジョブを要求したユーザ名や印刷を行ったアプリケーション名など、「ジョブの開始点」に関する情報は詳細に採取できる。しかしながら、最終的に何枚出力されたかなどの「ジョブの終了点」に関する情報や、印刷サーバが行う予定の印刷付加価値ジョブスケジューリングに関する情報を保管することができない。

【0008】また印刷サーバは、印刷サーバ上のジョブ

スケジューリング、分散印刷、正常に動作しない印刷装置回避による別の印刷装置への出力等、印刷サーバがサービスする情報に関しては詳細に採取可能である。しかしながら、印刷サーバは、ジョブ開始点や終了点に関する情報を保管できない。

【0009】さらに印刷装置は、実際に出力された枚数など、ジョブの終了点に関する情報は詳細に採取できるが、要求者などの開始点の情報や、サーバによる印刷付加価値情報などは保管できない。

【0010】印刷ジョブは開始点からサービス点を通して、終了点に到達するが、その間の各点で取得し得る印刷ジョブ情報は異なっており、従来技術によって生成されるジョブ履歴情報は、このパス上のいずれか一点のジョブ情報に偏っているという問題点があった。

【0011】また、ジョブ履歴は、課金、負荷分析など様々な用途で用いられる。必要なジョブ情報は用途によって異なっており、場合によってジョブのパス上での全情報が必要になる場合もある。上記されたような問題点があると、ジョブ情報を使用する上で不便な場合があった。

【0012】本発明は、上記問題点を鑑みてなされたものであり、印刷ジョブに関する一連の情報を統合化し、統合化されたジョブの全体像をユーザに提供する印刷システムを提供することを目的とする。

【0013】さらに、1つの印刷ジョブの生成から消滅までに与えられる識別情報を印刷システム上の各々のサブシステムが取得するものを提供することも目的とする。また、ジョブ情報断片が増えなくなり、情報をクローズするタイミングを認識するものを提供することを目的とする。

【0014】

【課題を解決するための手段】かかる目的を達成するため、請求項1に記載の発明は、ネットワーク上のクライアントが有する印刷データ生成部が生成する印刷データを印刷装置駆動部がネットワークを通じて送信して、送信された印刷データを印刷装置で印刷する印刷ジョブを行い、印刷管理装置が接続された印刷システムであって、印刷装置駆動部は、印刷ジョブにおいて印刷装置駆動部がどのような処理を行ったかというドライバ履歴情報を記憶し、保存する第1の記憶手段を有し、印刷装置は、印刷ジョブにおいて印刷装置がどのような処理を行ったかという印刷装置履歴情報を記憶し、保存する第2の記憶手段を有し、印刷管理装置は、第1の記憶手段が記憶するドライバ履歴情報と、第2の記憶手段が記憶する印刷装置履歴情報とを読み出す読み出し手段と、印刷管理装置が有するデータを表示する表示手段とを有することを特徴とする。

【0015】請求項2に記載の発明は、請求項1に記載の発明において、印刷管理装置は、読み出し手段によって読み出されたドライバ履歴情報および印刷装置履歴情

報を記憶し、保存する第3の記憶手段をさらに有することを特徴とする。

【0016】請求項3に記載の発明は、請求項1または2に記載の発明において、印刷管理装置は、読み出し手段によって読み出されたドライバ履歴情報および印刷装置履歴情報のいずれか1つに含まれる任意の1つのデータに示される処理を選択し、基準の処理を決定する基準ジョブ決定手段と、基準ジョブ決定手段によって決定された基準の処理とは別のデータで示される処理が同一の印刷ジョブで行われているか否かを判断する判断手段と、基準の処理を示すデータとは別の処理を示すデータを、基準の処理を示すデータに追加するデータ追加手段と、基準の処理とは別のデータで示される処理が判断手段によって同一の印刷ジョブで行われていると判断された場合、データ追加手段によって追加されることを特徴とする。

【0017】請求項4に記載の発明は、請求項3に記載の発明において、追加手段によって追加された別の処理を示すデータを、追加手段によって追加される前に格納されていたドライバ履歴情報および印刷装置履歴情報のいずれか1つから削除する第1の削除手段を有することを特徴とする。

【0018】請求項5に記載の発明は、請求項3または4に記載の発明において、印刷データ生成部は、ネットワーク上およびネットワークに接続される端末内で共通で、各々の印刷ジョブによって区別される識別情報を印刷データに組み込む識別情報組み込み手段を有し、判断手段による判断の基準は、識別情報組み込み手段によって組み込まれた識別情報に基づくことを特徴とする。

【0019】請求項6に記載の発明は、請求項1から5のいずれか1項に記載の発明において、印刷管理装置は、ドライバ履歴情報および印刷装置履歴情報のいずれか1つに含まれる1つの情報を構成する1つの要素である情報項目を削除する第2の削除手段を有し、判断手段によって判断される際に、ドライバ履歴情報の情報項目と印刷装置履歴情報の情報項目とが一致する場合、ドライバ履歴情報の情報項目および印刷装置履歴情報の情報項目のいずれか一方を第2の削除手段によって削除することを特徴とする。

【0020】請求項7に記載の発明は、請求項1に記載の発明において、ネットワークにはさらにサーバが接続され、サーバは印刷ジョブにおいてサーバがどのような処理を行ったかというサーバ履歴情報を記憶し、保存する第4の記憶手段を有し、読み出し手段は、第4の記憶手段が記憶するサーバ履歴情報をさらに読み出すことを特徴とする。

【0021】請求項8に記載の発明は、請求項2に記載の発明において、ネットワークにはさらにサーバが接続され、第3の記憶手段はサーバ履歴情報をさらに記憶することを特徴とする。

【0022】請求項9に記載の発明は、請求項3または5に記載の発明において、ネットワークにはさらにサーバが接続され、基準ジョブ決定手段の選択の対象はサーバ履歴情報に含まれる任意の1つのデータに示される処理も含まれることを特徴とする。

【0023】請求項10に記載の発明は、請求項4に記載の発明において、ネットワークにはさらにサーバが接続され、第1の削除手段の削除の対象はサーバ履歴情報も含まれることを特徴とする。

【0024】請求項11に記載の発明は、請求項6に記載の発明において、ネットワークにはさらにサーバが接続され、第2の削除手段による削除の対象はサーバ履歴情報に含まれる1つのデータを構成する1つの要素である情報項目も含まれることを特徴とする。

【0025】請求項12記載の印刷システムの発明は、複数のサブシステムを有する印刷システムであって、印刷システムは、印刷システムで行われる各々の印刷ジョブを識別するジョブ識別子と、印刷システムの所定の範囲内でジョブ識別子を検索するために用いられるキーとを対応づけて保存するジョブID一覧保持部をさらに有し、複数のサブシステムは、キーによって識別子を検索し、識別子によって、印刷ジョブにおいて複数のサブシステム各々で行われるジョブである断片ジョブをまとめることを特徴とする。

【0026】請求項13記載の発明は、請求項12の印刷システムにおいて、ジョブID一覧保持部は、ジョブ識別子を破棄する識別子破棄手段と、破棄する時期を決める破棄時期決定手段と、をさらに有し、破棄する時期は複数のサブシステムの通知によって決定されることを特徴とする。

【0027】請求項14記載の発明は、請求項12または13の印刷システムにおいて、複数のサブシステムは、サブシステムにおける断片ジョブの開始と終了とをジョブID一覧保持部に通知するジョブ開始終了通知手段をさらに有し、サブシステムは、ジョブ開始終了通知手段が断片ジョブの終了を通知してから、断片ジョブをまとめることを特徴とする。

【0028】

【発明の実施の形態】以下に添付図面を参照した本発明に係る印刷システムの実施形態を詳細に説明する。図1から図10に、本発明に係る印刷システムの一実施例が示す。

【0029】[第1の実施例] 図1に、本発明に係る印刷システムの第1の実施例を示す。図1に示すように、本実施例は、クライアント10、サーバPC20、印刷装置30およびジョブ情報履歴装置40がネットワークを介して接続されて構成されている。

【0030】以下に、一般的な印刷ジョブの流れを示す。まずアプリケーション11で印刷データの生成が指示されて、ドライバ12で印刷データが生成する。生成

した印刷データは、サーバ通信13によってサーバPC 20の付加サービス21に送信される。付加サービス21によって分散印刷、異常プリンタ回避などの付加サービスが提供され、印刷装置30において印刷データが印刷される。

【0031】上記された印刷ジョブにおいて、ドライバ12、サーバ通信13、付加サービス21および印刷装置30のそれぞれにおいて、行ったジョブの履歴が蓄積される。しかしながら、上記された各モジュール（ドライバ12、サーバ通信13等）に保存される履歴情報だけでは、一連の印刷ジョブの全体的な流れで各モジュールが行った動作を把握できるように、以下のように行う。ここで本実施例では、図1に示されるように、各々のモジュールが保存する履歴情報を統合し、管理するジョブ情報履歴装置40がネットワークに接続されている。

【0032】以下では図4に基づいて、ジョブ情報履歴装置40が各モジュールの履歴情報を収集し、一連の印刷ジョブの流れを示す情報が作成される。

【0033】ジョブ情報履歴装置40は、図2に示されるような各モジュールが生成したジョブ履歴情報を収集する（ステップS1）。収集された複数のジョブ履歴情報の内、一つの履歴情報を選択して、選択された履歴情報を基準履歴情報とする（ステップS2）。基準履歴情報に含まれるジョブ情報を全部走査する（ステップS3）。走査されたジョブの内の任意の1つを基準ジョブとし（ステップS4）、以下このジョブが行われた印刷ジョブの他のモジュールに格納される履歴が収集される。

【0034】マージの対象となる任意の履歴情報を1つ取り出す（ステップS5）。基準ジョブと同一の印刷データを印刷する際に行われたと判断されるジョブを残りの各履歴情報の中から抽出し（ステップS6）、基準ジョブのジョブ情報に追加する（ステップS7）。本実施例では判断の基準は印刷が行われた日時と印刷データのサイズに基づいている。

【0035】抽出されたジョブ情報を履歴情報から削除する（ステップS8）、追加したジョブ情報でその印刷ジョブの全情報がマージされていなければ（ステップS9/No）、別の履歴情報を取り出す（ステップS5）。すべての履歴情報がマージされていれば（ステップS9/Yes）、取得したすべてのジョブ情報を1つの印刷ジョブのジョブ情報として統合化して、先に統合化されている統合化履歴情報に追加する（ステップS10）。

【0036】基準履歴情報中のすべてのジョブが基準ジョブになっていなければ（ステップS11/No）、別の任意のジョブを基準ジョブにする（ステップS4）。基準履歴情報中のすべてのジョブが基準ジョブになっており（ステップS11/Yes）、かつ、収集した履歴

情報が基準履歴情報になっていれば（ステップS12/Yes）、処理は終了する。収集した履歴情報がすべて基準履歴情報になっていなければ（ステップS12/No）、別の履歴情報が基準履歴情報になる（ステップS2）。

【0037】上記されたように1回の印刷ジョブについて、各モジュールに保存されるジョブ情報が統合化することによって、例えば、その結果を印刷するなどしてユーザが入手すれば、ユーザは従来に比べてそれぞれの印刷ジョブについて全体的な情報を入手することが容易になる。

【0038】本実施例では、ジョブ情報履歴装置40は各モジュールの生成した履歴情報を統合化する時に、ある基準履歴情報中の基準ジョブと同一の印刷ジョブの際に行われたと判断されるジョブ情報を、残りの各履歴情報中から取得する際に判断基準が使用される。この判断基準は、その印刷ジョブが行われた日時とサイズの一致に基づいて行われる。

【0039】〔第2の実施例〕以下に、第2の実施例を示す。本実施例では、上記第1の実施例において、上記された判断基準に基づいて判断された場合、同一の日時に同一のサイズの印刷ジョブが行われた時、別の印刷処理のジョブであったとしても、同一の印刷ジョブと判断してしまう可能性がある場合の回避例を示す。これ以下、第2の実施例としてPJL等の印刷制御情報中に「JOBID=識別情報」を埋め込む方法について説明する。図5に、各モジュールが保存する履歴情報の内容を示す全体の構成図が示されている。

【0040】本実施例では、ドライバ12等の印刷ジョブの開始点となるモジュールが、印刷データ中にジョブの識別情報を埋め込む。この識別情報は、システムやネットワーク全体の中で、その印刷ジョブであると特定できるものであるとする。識別情報を埋め込む方法としては、上記されたようにPJL等の印刷制御情報中に「JOBID=識別情報」を付加する方法が選択される。

【0041】以下、図7(a)に示されるフローチャートに基づいて、本実施例におけるドライバ12等のジョブの開始点において、識別情報をジョブデータに埋め込む処理を説明する。アプリケーションやシステムから、印刷要求の開始と印刷データの生成が依頼される（ステップS11）。

【0042】印刷データとジョブの識別情報とを生成し（ステップS12）、生成した印刷データとジョブの識別情報とをセットにして履歴情報に保管し（ステップS13）、ジョブの識別情報をジョブデータに埋め込み、その後、印刷工程（付加サービス等）に送る（ステップS14）。

【0043】以下、図7(b)に示すフローチャートに基づいて、本実施例におけるジョブの開始点以外の点における処理を説明する。モジュールの前の工程から送信

されたジョブデータを受け取り（ステップS21）、ジョブデータに従って各モジュールの一般の処理（付加サービス21ならサーバサービスで、印刷装置30なら紙出力）を行う（ステップS22）。

【0044】ジョブデータ中のジョブ識別情報を抽出し、ジョブ識別情報とその他の情報（処理結果の情報やジョブ情報など）とをセットにして履歴情報に保管する（ステップS23）。上記各モジュールが保存したジョブ履歴情報を、図6に示す。

【0045】各モジュールが保存したジョブ履歴情報群を統合するときは、基本的な統合化処理の流れは前記第1の実施例の場合と同様である。第1の実施例では、基準ジョブと同じ印刷ジョブで行われたジョブを他の履歴情報中から検索する場合は、日時とサイズの一致を判断基準としていた。本実施例では各ジョブの識別情報の一致により検索を実行する。

【0046】システム全体で印刷ジョブを特定できるジョブ識別情報の生成に関しては、クライアント10のネットワークアドレスを使用する方法などが考えられる。ネットワークアドレスを用いることによりシステム上のネットワークノードが特定されるため、識別情報の前半にクライアントのネットワークアドレスあるいはそれに基づいた値を用いる。

【0047】識別情報の後半には、そのネットワークノード（具体的にはクライアント10）内で特有の値を付加する。例えば、特有の値としては、ミリ秒やマイクロ秒レベルの時間情報、あるいは、インクリメント・カウンタなどが考えられる。さらに分散コンピューティング技術におけるいくつかの「ユニーク値生成方式」（UUIDなど）を用いてもよい。

【0048】また、ドライバ12等の印刷データを生成するモジュールが、ジョブ識別情報を埋め込めないシステム構成の場合は、その後の工程モジュールが、前工程モジュールから送られた印刷データに対し、後から識別情報を付加する方法が採用される。

【0049】また印刷データの形式によっては識別情報を埋め込むことができない場合もあるが、その場合は前工程から後工程に印刷データを渡す通信手段の中で、識別情報を送信してもよい。つまり印刷データ内に識別情報を埋め込むのではなく、パケットのヘッダやコマンドの引数など、データ受け渡し手段の方を経由してジョブ識別情報を送信してもよい。

【0050】また通信手段によっては、ジョブを送信するのではなく、受信側が識別情報を付加する場合には、前工程（ジョブを送信する側）が後工程（ジョブを受け取る側）からその識別情報を受け取って、履歴情報に追加してもよい。

【0051】上記した本実施例に示すように、印刷ジョブに識別情報を新たに埋め込むことによって、印刷ジョブ情報を統合化する際に、違う印刷ジョブで行われたジ

ョブであっても同じ印刷ジョブで行われたジョブと判断される可能性がなくなる。これにより、ユーザはより正確なジョブ情報を把握することができるようになる。

【0052】[第3の実施例]以下、第1の実施例において、文書名（ジョブ名）、要求者情報や課金情報など、ジョブが開始されてから完了するまで変化しないと予想される情報が同一印刷ジョブの複数の履歴情報中で重複して現れる場合の、重複する情報を1つにマージして情報を統合化する方法について、第3の実施例として説明する。図8に、第3の実施例における全体の構成図を示す。図8において、各モジュールが保存する履歴情報の内容も示す。

【0053】以下に、図10に示すフローチャートに基づいて、本実施例における動作を示す。以下の動作は図4に示すフローチャートのS6とS7との間で行われる。まず基準ジョブの中から情報項目の項目識別子をすべて取得する（ステップS31）。ステップS6において抽出された統合化対象のジョブ中で、S31で取得した項目識別子と重なるものを削除する（ステップS32）。

【0054】この第3の実施例では、例としてジョブ情報は、ジョブ識別情報、日付、ユーザ名およびバイト数の4つから構成されるものとする。これら4つの情報は、ジョブ情報を表現する個々の構成要素であると考えられ、これらの構成要素は、以下に示すように、「タイプと値」の組から構成される。

【0055】例えば、ジョブ情報が上記4つの上記4つの要素から構成される場合は、1つのジョブ情報は「"jobid=1234"」「"date=06/01"」「"user=浅谷"」「"size=180"」という4つの「タイプ+値」組から成る。

【0056】各履歴情報は各モジュールにおけるジョブの履歴を並べたもので、その個々のジョブ情報に、その値だけでなく、項目識別情報も含まれる。項目識別情報は情報要素のタイプを意味し、具体的には日時を表す情報の場合には日時情報を、バイト数を表す情報の場合にはバイト数情報のタイプである。

【0057】履歴情報がCSV（comma separated value）形式のデータである場合、

```
jobid,    date,    user
1234,    06/01, 浅谷
1235,    06/01, 井上
```

の様になり、一行目にどのようなタイプのデータがどの順で並んでいるかを表す行を設けて識別するという方法がある。

【0058】あるいは履歴情報をXML（extensible markup language）などのタグ付け形式にして、各情報要素が異なるタグ名で記載される方法もある。また履歴情報がバイナリデータの場合、各情報要素はその値だけではなく「タイプ、値」の列で格納される方法もある。

【0059】履歴情報を生成する各モジュールは、同じ意味の情報要素については同じ情報識別子で識別するようにする。例えば「ジョブ識別情報」という情報要素は、同じ印刷ジョブで行われている以上、ジョブのパスの開始から終了のいかなる点で採取しても同じ意味を持つ。

【0060】システムによっては、ジョブ名やジョブの要求者名も、ジョブの開始から終了まで同じ意味を持つ場合がある。上記されたような情報要素については、すべてのログ情報保存モジュールが、jobid、jobname あるいはuserなどの同一の情報要素識別子を用いるようにする。

【0061】上記されたように型が付けられた履歴情報において、これらを統合化する場合には、同じ情報要素であれば一つにまとめることが可能になる。その処理の流れを図10に示す。以下、まとめる処理について、図10に基づいて説明する。

【0062】第1の実施例において、基準ジョブに別の履歴情報中の同一の印刷ジョブで行われたと判断されるジョブの情報を追加する時に、単に追加するのではなく、基準ジョブ情報内に既に存在する情報要素を1つ列挙する。そして各々の既存情報要素に対して、情報要素識別子をもつ情報要素を探し、一致する情報要素があれば、それを削除する。こうして残された情報要素だけが基準ジョブ情報に追加される。

【0063】本実施例では、2つのジョブ情報の中に重複する情報要素があった時に、基準ジョブ情報の値を優先する方法であった。しかしながら、基準ジョブ情報の値を統合化対象のジョブ情報の値で上書きしてもよい。また、双方の値を比較して、一致しない場合には、ユーザに一方の値を選択させたり、一致しなかった場合には、あらかじめユーザがどちらの値を選ぶかを指定できるようにするという方法を採用する。

【0064】上記されたように、複数の履歴情報に存在する同一印刷ジョブで行われたジョブの情報が統合化される際に、重複する情報要素を検知しマージすることで、統合化されたジョブ履歴情報は、情報を失わずに簡素化され、ユーザがデータを閲覧する際にも見易くなる。

【0065】〔第4の実施例〕ジョブ識別情報に関する上記の説明では、印刷ジョブの情報断片に識別情報を必ず埋め込むものとして説明したが、ジョブのデータ形式によっては、埋め込むことができない場合がある。また、ジョブ情報断片を供給するサブシステムの中には、ジョブデータの中身を見ることができない場合がある。

【0066】また、サブシステム間のジョブ搬送手段は、ジョブデータを埋め込まない場合もあり、さらには、ジョブ情報断片を供給するサブシステムの中にジョブの搬送に関与しないものもある。

【0067】以下、図11の本発明の一実施形態として

適用される印刷システムの構成図に基づいて、キーによって識別子を得る処理について説明する。ジョブの識別子を採番したサブシステムは、その識別子と、そのジョブを検索し得るキーのペアを、ジョブID一覧保持部に登録する。一方、ジョブ情報断片を供給するサブシステムは、そのジョブを検索し得るキーをジョブID一覧保持部に渡して、対応する識別子を受け取る。

【0068】以下、「ジョブを検索し得るキー」と「ジョブを識別する識別子」との違いについて説明する。まず、「ジョブを識別する識別子」とは、ジョブの生成から消滅までの時間、システム全体でジョブを一意に識別するようなものである。

【0069】一般に印刷を行うシステムでは、ジョブは例えばClient PC上で生成され、PC上の様々なモジュールを通り、やがて、そのジョブはPCの外に出て、印刷サーバに渡され、サーバPC内の様々なモジュールを通り、さらに、サーバPCを出て印刷装置に渡される。

【0070】他方、キーとは範囲の限定されたコンテキスト内で、ジョブを識別可能な値である。例えば、Client PC上でOSが番号をふる、すなわち採番するジョブIDは、そのPC内に限定すれば、ジョブを識別できる。しかし、この値は、ジョブが印刷サーバや印刷装置など、Client PC外ではジョブの識別に用いることができない。また、OSによっては、再起動するだけでジョブ識別ができなくなるものもある。この場合、Client PC上でOSが番号をふるIDは上記キーである。

【0071】ジョブID一覧保持部がこのClient PC上で稼働する場合、ジョブID一覧保持部は、OSが番号を振ったIDであるキーとジョブ識別子とのペアを保持する。Client PC上でジョブ情報を供給する様々なモジュール（例えば、印刷アプリケーションや、プリンタドライバ、ポートドライバ、印刷プロセスなど）は、OSのジョブIDから、システム全体に渡って有効なジョブ識別子入手することができる。

【0072】以下、サブシステム間でジョブを搬送する例について説明する。例えば、ネットワーク印刷プロトコルの中には、ジョブ受信側（印刷サーバなど）が、ジョブを受け取るときにふったIDをジョブ送出側（印刷クライアントなど）に渡すものがある。

【0073】この場合、IDとしてどのような値を用いるかの決定権は、ジョブ受信側にある。本来、ジョブの生成時に識別子を生成し、ジョブの生成と同じ時点からのそのジョブの識別を可能にしなければ、ジョブの生成初期の時点で関与するサブシステムはジョブ情報を生成できない（生成できても、識別子は埋め込めない）。

【0074】また、ジョブ受信側の中には、このジョブ受信側のみでジョブを識別可能な値、すなわち、他のサブシステムでは識別可能な値を返すものもある。また、印刷プロトコルによっては、ジョブ受信側が返すジョブ

IDのデータ形式が決められている場合がある。この場合、システム全体でジョブを識別するための共通の情報を埋め込むことができない。

【0075】従って、システム全般でジョブを識別可能な識別子は、別途に採番される。例えば、ジョブの生成時点で番号をふられるようにしてもよい。そして、この例では、キーは印刷サーバなどが生成するIDとなる。

【0076】この場合、「ClientなどのサブシステムAと、サーバなどのサブシステムBの間のジョブ搬送」という限定されたコンテキストで、搬送プロトコル上のIDは有用なものである。従って、この間の授受に対してジョブID一覧保持部を起し、ジョブ送出側と受信側の二者間で有効なIDをキーとして、ジョブフローの上流から引き継がれてきたジョブ識別子を検索可能にする。

【0077】このように既存の技術で導入されているジョブID値とは、Client PC内のOS範囲内や、二者間のジョブ搬送通信など、限定されたコンテキストで有用な値である。既存技術ではこのようなIDは多いので、それをそのコンテキスト内における「キー」とし、そのキーからシステム全体でジョブを識別できる識別子を検索する。

【0078】なお、本例では、「ジョブID一覧保持部をClient PC内に置き、Client PC内の各モジュールでのジョブ識別子検索を可能にする」「ジョブ送信側と受信側の二者間にジョブID一覧保持部を置き、この二者間でのジョブ識別子検索を可能にする」という方法を例示した。この例は、ジョブID一覧保持部を、各々のコンテキスト内に置くモデルである。

【0079】しかし、別の方法として、ジョブID一覧保持部を各々のコンテキストの外に置き、システム全体で活用できる位置付けとすることもできる。そのためには、例えば、[key-識別子]というペアの配列でなく、[context-Aのkey, context-Bのkey, context-Cのkey-識別子]というように、一つのジョブ識別子に対して、各コンテキスト毎のkeyを結び付けられるようにすればよい。

【0080】以下、図12(a)および(b)に基づいてジョブ生成側サブシステムおよびジョブ受信/監視側サブシステムが行う処理について説明する。まず、ジョブ生成側サブシステムがジョブID一覧保管部にキーと識別子のペアを保管する処理について説明する。

【0081】ジョブ識別子を生成する(ステップS41)と、そのジョブを限定されたコンテキスト内であれば別の方法で特定できるようなキーを決定する(ステップS42)。ジョブ識別子とキーとを、ジョブID一覧保持部に通知する(ステップS43)。ジョブID一覧保持部は、キーと識別子とのペアを保管して(ステップS44)、処理が終了する。

【0082】次に、ジョブ受信/監視側サブシステムがキーと一致する識別子を返す処理について説明する。ジ

ョブ情報を取得すると(ステップS51)、そのジョブを限定されたコンテキスト内であれば別の方法で特定できるようなキーを決定する(ステップS52)。

【0083】キーをジョブID一覧保持部に送り、対応する識別子を要求する(ステップS53)。ジョブID一覧保持部はID一覧を検索し、キーと一致する識別子を返す(ステップS54)。以下、消去する時期を認識する処理について説明する。

【0084】ジョブ識別子一覧は各サブシステムが保持する。各サブシステムは、自分がジョブを処理する過程で、必要なジョブ識別子のみ記憶し、ジョブ情報を送出し終えた場合に不要となった識別子が発生する。この識別子は、サブシステム自身の適切なタイミングで破棄すればよい。

【0085】ID通知部が[key]-[識別子]の情報を各サブシステムに通知する手段としては、予めそのような情報を必要とするサブシステムが自身を通知先として登録する方法もあるし、あるいは、ネットワーク上のbroadcast/multicastのようにあらかじめ、決められた手段で情報を不特定多数に流し、各サブシステムが必要な分だけ拾うという方法もある。

【0086】以下、図15の(a)および(b)に基づいて、識別子の番号をふる処理(採番処理)について説明する。まず、(a)に基づいてこの処理を説明する。

【0087】アプリケーションやドライバなどジョブ識別子の番号をふるモジュールは、識別子とキーとをID通知部に渡す(ステップS61)。ID通知部はこのキーと識別子とを各サブシステムに通知する(ステップS62)。キーと識別子とを受信した各サブシステムは、それを自身で保持して(ステップS63)、処理は終了する。

【0088】次に、(b)に基づいてサブシステムによるジョブ情報生成時の処理について説明する。対象とするジョブのキーをもとにして、自身が持つテーブルから対応する識別子を取り出す(ステップS71)。そのジョブの情報を送出し、自身のテーブルからそのジョブの[キー]-[識別子]を削除する(ステップS72)。

【0089】以下、ジョブ履歴保持部が、任意のジョブ情報において、情報断片の受信待機時間の判断方法について説明する。ここで、ジョブが通るパス上のサブシステムは様々なタイミングでジョブ情報断片をジョブ履歴保持部に供給するものとする。

【0090】図16に、本発明の一実施形態として適用される印刷システムの構成図が示す。各サブシステムは、自身のもとにジョブが搬送された時に、開始を表す通知をジョブ履歴保持部αに送る。この時、この通知だけでなく、開始に関するジョブ情報断片と一緒に送ってもよい。

【0091】サブシステムはジョブ処理を行った後、次工程のサブシステムにジョブを搬送する。サブシステム

はジョブ処理を開始したら、Start の通知をジョブ履歴情報保持部に送る。前工程のサブシステムは、ジョブの転送が完了した後、ジョブ情報断片を送らないことを表す通知を送る。なお、前記同様に、自身のジョブ処理の終了に関するジョブ情報断片と一緒に送るようにしてもよい。

【0092】次工程のサブシステムも同様に、ジョブ処理を行った後に次工程に送り、Endの通知（必要ならばジョブ情報断片も）ジョブ履歴保持部に送る。サブシステムの中には、ジョブのフローに介在せず、ジョブパスの外から監視を行うものもある。本例では、印刷装置を監視するサブシステムがあるとする。

【0093】このサブシステムは、自身が監視するポイントでジョブが発生（あるいは受信）したことを発見した場合、前記同様にStart の通知や補足のジョブ情報断片をジョブ履歴保持部に送る。また、そのポイントにおいてジョブが消滅したら、End の通知や補足のジョブ情報断片をジョブ履歴保持部に送る。

【0094】一方、ジョブ履歴保持部は、Start - End やそれに付随するジョブ情報断片を蓄積する。任意のジョブに対し、一のサブシステムからEnd の通知が来ても、他のサブシステムからはStart の通知しかきていない場合、このジョブは、システム上で生存し、今後も情報断片が寄せられる可能性がある。その場合は、そのジョブに関しては、まだオープン状態とする。

【0095】Start を送ったすべてのサブシステムから、そのジョブに対するEnd の通知が来たら、システム上でそのジョブに対する情報断片はそれ以上寄せられないと判断し、そのジョブの情報をクローズし、情報断片のマージを行う。以上のようにすれば、システム上の様々なサブシステムが、様々なタイミングでジョブ情報断片を供給する場合でも、情報断片をまとめあげる正確なタイミングを得ることができる。

【0096】図17に、上記の処理についての処理の流れを示す。まず、Subsystem Aにジョブが搬送されると、開始を示す通知がジョブ履歴保持部に送られ、ジョブ処理が開始される。そのジョブ処理が終了したら次工程のサブシステムであるSubsystem Bにそのジョブは搬送される。

【0097】その搬送が終了すると、終了を示す通知をジョブ履歴保持部に送る。一方、ジョブが搬送されたSubsystem Bは開始を示す通知をジョブ履歴保持部に送り、搬送されたジョブの処理を開始する。

【0098】同様に、処理が終了したら次工程のサブシステムである印刷装置にジョブを搬送する。搬送が終了したら終了を示す通知をジョブ履歴保持部に送る。一方、ジョブが搬送された印刷装置では搬送が終了したら、印刷処理を行って処理を終了する。

【0099】また、ジョブのフローに参加しないSubsystem Xでは、Subsystem Bから印刷装置へのジョブの搬

送が開始したら、ジョブ履歴保持部へその開始を示す通知を行い、上記システムにおけるすべての処理が終了したら、その終了を示す通知をジョブ履歴保持部へ行う。

【0100】

【発明の効果】以上の説明から明らかなように、本発明によれば、1回1回の印刷ジョブについて、各モジュールに保存されるジョブ情報が統合化されるので、その結果を印刷するなどしてユーザが入手しやすくすれば、ユーザは従来に比べてそれぞれの印刷ジョブについての全体的な情報を入手することが容易になる。

【0101】さらに印刷ジョブに識別情報を新たに埋め込むことによって、印刷ジョブ情報を統合化する際に、違う印刷ジョブで行われたジョブであっても同じ印刷ジョブで行われたジョブと判断される可能性がなくなるので、ユーザはより正確なジョブ情報を把握することができるようになる。

【0102】さらに、複数の履歴情報に存在する同一印刷ジョブで行われたジョブの情報が統合化される際に、重複する情報要素を検知しマージすることで、統合化されたジョブ履歴情報は、情報を失わずに簡素化され、ユーザがデータを閲覧する際にも見易くなる。

【0103】また、請求項12記載の発明によれば、ジョブ識別子を直接読み出すことができなくても、検索して読み出すことができるので、確実に識別子を読み出すことができ、より正確に印刷ジョブをまとめることが可能になる。

【0104】請求項13記載の発明によれば、請求項12記載の発明において、ジョブ履歴保持部において必要がなくなった識別子を破棄することができるようになったので、より効率よくジョブ履歴保持部を識別子の記憶領域を使用することが可能になった。

【0105】請求項14記載の発明によれば、請求項12または13記載の発明において、ジョブをまとめるタイミングをサブシステムが把握することが可能になったので、各サブシステムはジョブをまとめる適切なタイミングのもとにジョブをまとめることが可能になった。

【図面の簡単な説明】

【図1】本発明の印刷システムに係る第1の実施例の全体構成図である。

【図2】ドライバ12に保存される履歴情報12a、付加サービス21に保存される履歴情報21aおよび印刷装置30に保存される30a各々の履歴情報の内容の一例である。

【図3】第1の実施例における処理動作を実行するプログラムの内容が示されている。

【図4】第1の実施例における動作が示されるフローチャートである。

【図5】本発明の印刷システムに係る第2の実施例の全体構成図である。

【図6】第2の実施例において各モジュールが保存する

履歴情報の一例が示されている。

【図7】第2の実施例において各モジュールの動作が示されるフローチャートである。

【図8】本発明の印刷システムに係る第3の実施例における全体構成図である。

【図9】第3の実施例における処理動作を実行するプログラムが示されている。

【図10】第3の実施例における動作が示されるフローチャートである。

【図11】本発明の一実施形態として適用される印刷システムの構成図である。

【図12】ジョブ生成側のサブシステムがキーと識別子を保管する処理およびジョブ受信/監視側のサブシステムがキーから識別子を検索する処理が示されたフローチャートである。

【図13】印刷ジョブの基本モデル、ジョブデータへの埋め込みおよびジョブ搬送手段でのID通知について示される図である。

【図14】本発明の一実施形態として適用される印刷システムの構成図である。

【図15】識別子の番号をふる処理とサブシステムによってジョブ情報の生成時に行われる処理とが示されたフローチャートである。

【図16】本発明の一実施形態として適用される印刷システムの構成図である。

【図17】ジョブ履歴保持部に各サブシステムがジョブの開始と終了とを示すフローチャートである。

【符号の説明】

10 クライアント

11 アプリケーション

12 ドライバ

12a ドライバに保存される履歴情報

12b ドライバに保存される履歴情報

12c ドライバに保存される履歴情報

13 サーバ通信

13a サーバ通信に保存される履歴情報

13b サーバ通信に保存される履歴情報

13c サーバ通信に保存される履歴情報

20 サーバPC

21 付加サービス

21a 付加サービスに保存される履歴情報

21b 付加サービスに保存される履歴情報

21c 付加サービスに保存される履歴情報

30 印刷装置

30a 印刷装置に保存される履歴情報

30b 印刷装置に保存される履歴情報

30c 印刷装置に保存される履歴情報

40 ジョブ情報履歴装置

40a ジョブ情報履歴装置に保存される履歴情報

40b ジョブ情報履歴装置に保存される履歴情報

40c ジョブ情報履歴装置に保存される履歴情報

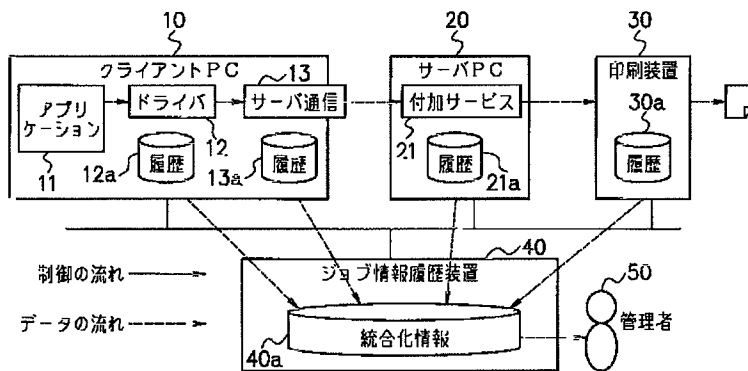
A サブシステム

B サブシステム

C サブシステム

X サブシステム

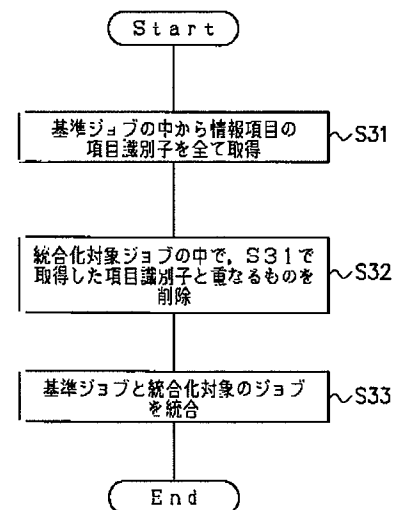
【図1】



【図2】

ドライバの履歴情報			サーバの履歴情報			印刷装置の履歴情報		
日時	バイト数	ユーザ名	日時	バイト数	時間	日時	バイト数	枚数
06/01	180	浅谷	06/01	180	10分	06/01	180	3
06/01	360	井上	06/01	360	50分	06/01	360	9
06/03	224	上野	06/03	224	4分	06/03	224	5

【図10】



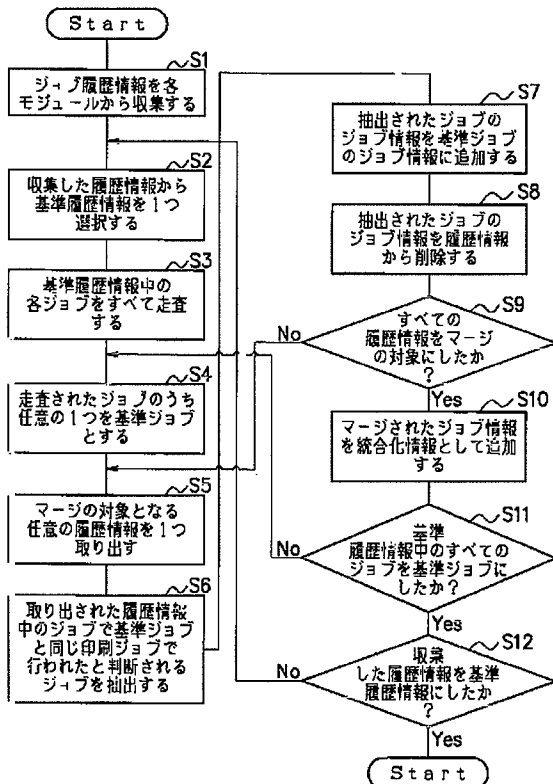
【図3】

```

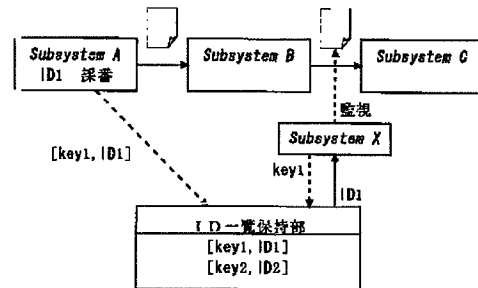
//history[]配列に三つの履歴情報が与えられたとする
//union_historyに統合化されたジョブ履歴情報を作成する処理の流れ
For baseidxをhistory[]の先頭ファイルから最終ファイルで{
    base_history = history[baseidx] //基準とする履歴ファイル
    For jobidxをbase_history内ジョブ一覧の先頭ジョブから最後まで{
        job = base_history[jobidx] //jobidx番目のジョブ情報
        For otheridxをhistory[]配列のbaseidxの次ファイルから最後まで{
            other_history = history[otheridx] //マージ対象の履歴ファイルを取り出す
            other_job = other_history[job, date, job, size] //別ファイル中の同じジョブの情報
            job.add_info(other_job) //ジョブ情報を追加
            other_history.delete_job(other_job) //重複検索を避けるためにジョブ情報削除
        }
        union_history.add_job(job) //マージされたジョブ情報を追加
    }
}

```

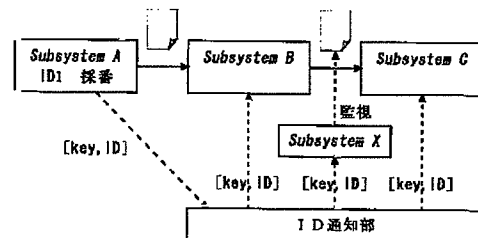
【図4】



【図11】



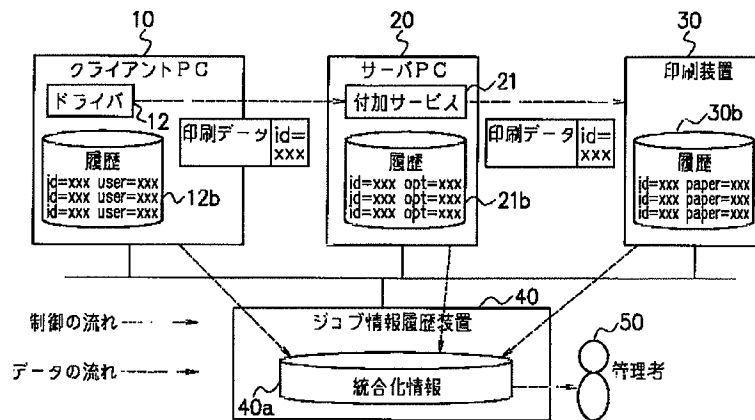
【図14】



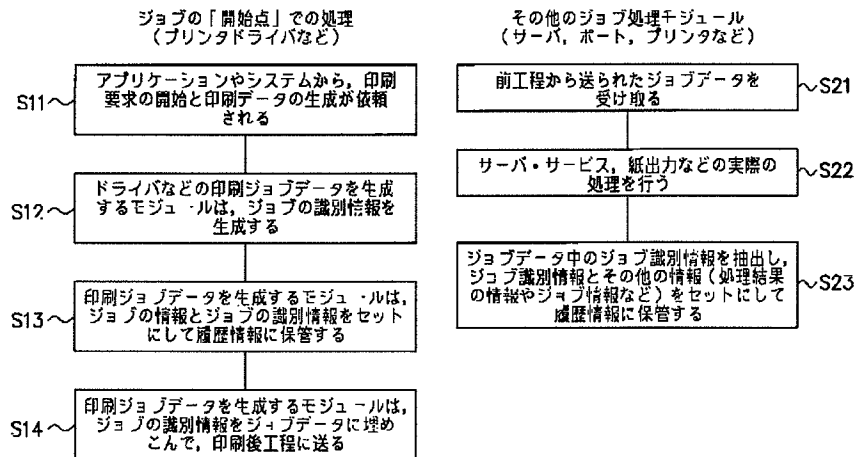
【図6】

識別情報	日時	サイズ	ユーザ	識別情報	日時	サイズ	時間	識別情報	日時	サイズ	枚数
001	06/01	180	浅谷	001	06/01	180	10分	001	06/01	180	3
002	06/01	360	井上	002	06/01	360	50分	002	06/01	360	9
003	06/03	224	上野	003	06/03	224	4分	003	06/03	224	5

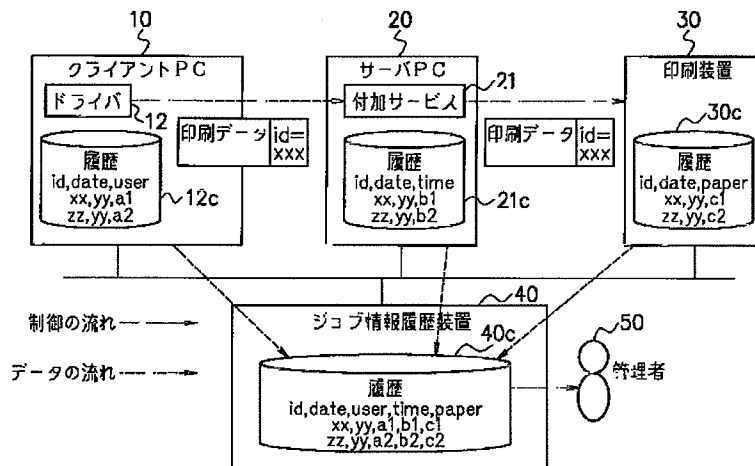
【図5】



【図7】



【図8】



【図9】

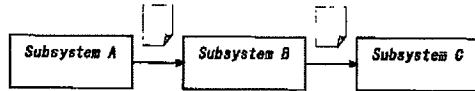
```

基準ジョブに対して他の履歴情報中の同じジョブの情報を追加するときに
//情報項目のマージを行う(基準ジョブをbase_job, 統合化対象のジョブをother_jobとする
add_info(base_job, other_job) {
  For Sitemをbase_job中の情報項目一覧の先頭から最後まで{
    itemtype = item.type //情報項目の項目識別子取得
    other_job.delete item(itemtype) //同じ項目識別子を持つ項目を削除
  }
  For Sitemをother_job中の情報項目一覧の先頭から最後まで{
    base_job.add item(item,name,item,value) //基準ジョブに項目識別子と値を追加
  }
}

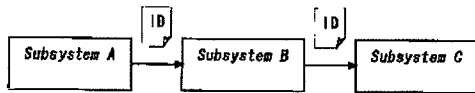
```

【図13】

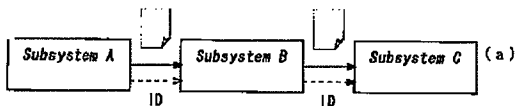
(a) 基本モデル



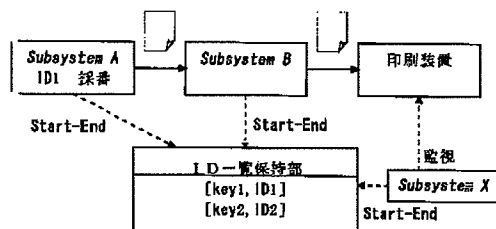
(b) ジョブデータへの埋め込み



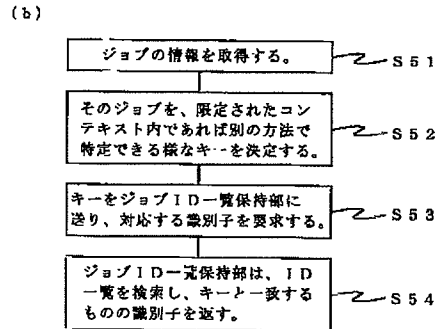
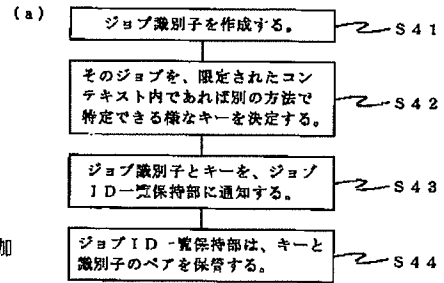
(c) ジョブ搬送手段としてのID通知



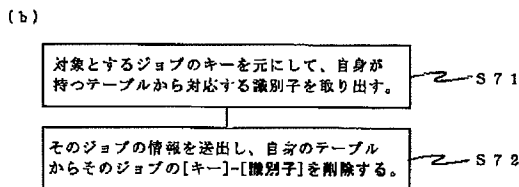
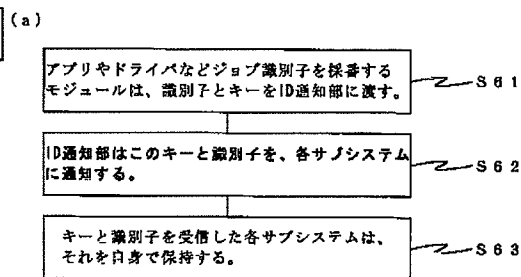
【図16】



【図12】



【図15】



【図17】

